NESTAR SYSTEMS, INCORPORATED

CLUSTER/ONE MODEL A (tm)

****
*
* Nestar Network Interface Card
*
* Theory of Operation
*
****

Publication Number ZA20-0105-0

INTRODUCTION

This document describes the internal operation of the Network Interface Card (NIC) used in the Nestar Cluster/One Model A system for the Apple][and Apple ///. It will explain the circuit diagram in detail and indicate how the various parts are used by the network software. The reader is assumed to be familiar with digital logic and microprocessor systems.

The Nestar part numbers for the NICs are A-2701, A-2702, or A-2703. As of this date the current PC board is 1015-2 Rev F (1/13/82), although this description with minor exceptions is applicable to all prior revisions of the 1015-2 board since Rev A (1/8/80).

General information about the network transmission scheme implemented by the NIC can be found in the article "Local Network Links Personal Computers" which appeared in Electronics, June 16, 1981, pps 171-175.

OVERALL STRUCTURE

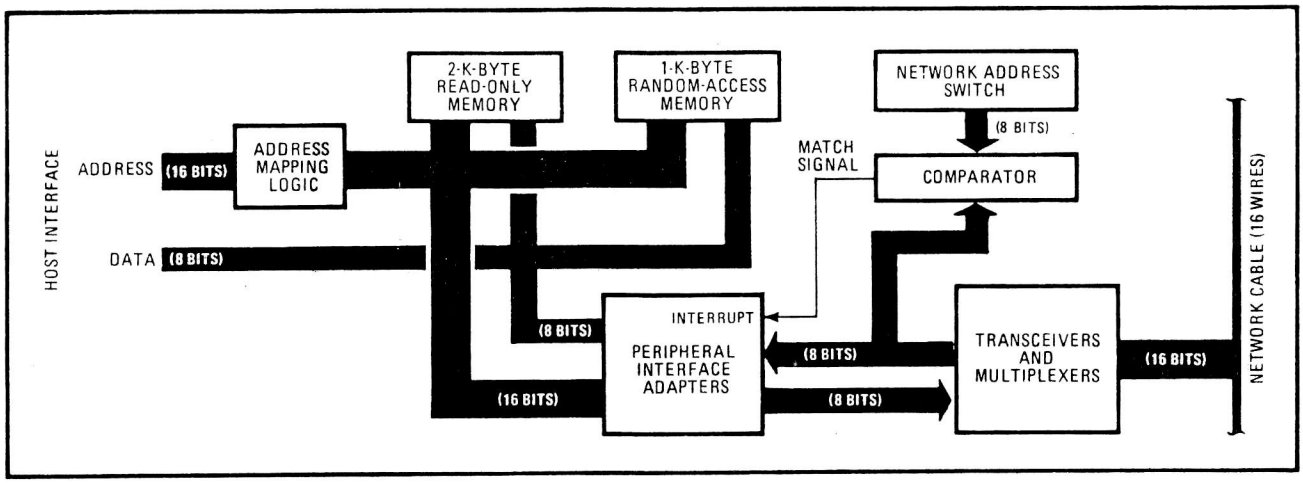
The NIC is a I/O peripheral card which obeys all the standards and conventions for the Apple][. Although the NIC was originally designed for the Apple][, it also works in the Apple /// in both native mode and Apple][emulation mode.

There are 5 major sections of the NIC:

- o The RAM: 1K bytes of local read/write memory
- o The PROM: 2K bytes of local program storage
- o The network interface: Logic and drivers to access the network
- o The address map: Decoding logic to map the RAM, PROM, and network interface into the Apple's address space
- o The timer: A 10 Hz timer for software timeouts

Figure 1 shows a block diagram of the NIC. The Apple address and data bus on the left enter from the edge connector which plugs into the Apple motherboard. The network connectors on the right are two 16-wire flat cable headers wired in parallel. The two connectors are identical and are convenient for daisy-chaining the network cable.

Figure 1 -- NIC block diagram



THE ADDRESS MAP

The Apple peripheral addressing convention

The 6502 processor on which the Apple is based has no explicit I/O instructions. All I/O references are made in memory-mapped fashion, and peripherals are designed to respond to specific memory addresses. The Apple peripheral card convention provides for three separate blocks of addresses that a peripheral may use:

16 bytes (DEVICE SELECT): 16 bytes starting at location C0n0, where n is the slot number plus 8. (The "slot" is the peripheral card connector, numbered from 0 to 7, into which the card is plugged.)

256 bytes (I/O SELECT): 256 bytes starting at location Cn00, where n is the slot number. This is often called the "slot-dependent memory page". It is not available to slot 0.

2K bytes (I/O STROBE): 2048 bytes starting at location C800.

The NIC does not use the 16-byte DEVICE SELECT area, but it does use both the 256-byte I/O SELECT and the 2K-byte I/O STROBE areas.

Note that the 2K address block is the same for all slots and may therefore be shared by several peripheral cards. Whenever the processor references a location in that 2K block, only the peripheral card which currently "owns" it is allowed to respond. The rules for ownership of the 2K block are as follows:

1. Any reference to the 256-byte I/O SELECT area of a peripheral slot gives ownership of the 2K block to that slot. The peripheral card typically sets a flipflop when I/O SELECT is active to record that it is the owner of the 2K block and may respond to the C800 addresses.

2. A reference to location CFFF causes no slot to own the 2K block. All peripheral cards typically reset their "ownership" flipflop whenever CFFF is read or written.

A typical programming sequence to "select a card" -- that is, to give ownership of the 2K block to a card in a particular slot -- is

```
LDA CFFF      ; deselect all cards
LDA C600      ; select the card in slot 6
```

Note that it requires cooperation from the software to insure that at most one card is selected. If more than one card is selected, as for example by the instructions

```
LDA C600      ; select slot 6
LDA C500      ; select slot 5 also!
```

then a reference to the 2K block will cause both cards to respond and there will be a data bus conflict which produces unpredictable results and may even cause damage to bus drivers.

The NIC address map

The NIC contains 2K of PROM, 1K of RAM, and 8 locations of PIA (Programmable Interface Adapter) registers which must be accessible by the 6502 processor. Since the sum exceeds the number of locations available to a single peripheral card (16+256+2048), a memory banking scheme is used.

Figure 2 shows the memory map of the NIC. The PROM is considered to be divided into 8 256-byte pages, and the RAM into 4 pages.

The slot-dependent 256-byte address space at Cn00 always accesses page 0 of the PROM and causes the card to be selected, that is, it then "owns" the 2K block. The first page of the 2K block (C8xx) is always page 0 of the RAM once the card is selected.

The next three pages are either the next three pages of the RAM, or the next three pages of the ROM, depending on whether the NIC is is "RAM mode" or "ROM mode". The NIC is initially in ROM mode after a reset has occured. The mode is set by one of the outputs of the PIA and can be changed by the processor in a manner explained later.

The last 4 pages of the 2K space are always the last 4 pages of the PROM, with the exception of the the last 16 bytes. Eight of those 16 bytes provide access to the PIA registers, and the other eight, ending in CFFF, are all locations which will deselect the

NIC although typically only CFFF is ever used.

Note that when location CFFF is referenced, the NIC is deselected after the reference is complete. Thus a load from location CFFF will load the last byte of the PROM, and then deselect the card. The NIC PROM in fact has a return (RTS) instruction as the last byte, so that a JMP CFFF will cause a subroutine return and deselect the NIC.

Figure 2 -- NIC Memory map

Block	Address	If "ROM mode"	If "RAM mode"
I/O SELECT	CnOO-CnFF	Page 0 of ROM	<--- same
I/O STROBE	C800-C8FF	Page 0 of RAM	<--- same
	C900-C9FF	Page 1 of ROM	Page 1 of RAM
	CA00-CAFF	Page 2 of ROM	Page 2 of RAM
	CB00-CBFF	Page 3 of ROM	Page 3 of RAM
	CC00-CCFF	Page 4 of ROM	<--- same
	CD00-CDFF	Page 5 of ROM	<--- same
	CE00-CEFF	Page 6 of ROM	<--- same
	CF00-CFEF	Page 7 of ROM (except last 16)	<--- same
	CFF0-CFF3	PIA #1 registers	<--- same
	CFF4-CFF7	PIA #2 registers	<--- same
	CFF8-CFFF	2K block deselect and last 8 bytes of Page 7 of ROM	<--- same

The sections which follow reference component designators on the NIC schematic, which is figure 6.

NIC Address Decoding

The heart of the NIC address decoder, which implements the address mapping described in the previous section, is the 256 x 4 PROM at location U14. It takes as input some single address lines, some address line combinations formed with external gates, and the status signals ENB (indicating that this card is selected) and ROM MODE (indicating ROM mode and not RAM mode). It is programmed with a table that then generates one of four select signals:

ROM SEL to indicate that the ROM is being read

RAM SEL to indicate that the RAM is being accessed. It is gated with the system clock to provide a clean chip select to the RAMs.

PIA SEL to indicate that one of the PIAs is being accessed.

DISABLE to indicate that a deselect reference (CFF8-CFFF) is being made.

One of the reasons for this decoding approach, which doesn't use the I/O STROBE signal to indicate a 2K block address reference, is due to the timing requirements of the PIAs. The PIA clock ("E") is the Phi-0 system clock, which is also the clock which strobes all the Apple select signals, including I/O STROBE. If I/O STROBE were included as an input to the circuit which generates the select signals (CS0, CS1, or CS2) for the PIAs, then the setup time for those signals with respect to the PIA clock cannot be met.

The gates U11-6 and U15-8 combine to decode any Cxxx reference if this card is selected. When that occurs, the address decode PROM and the bus buffer U16 are powered on by transistor switch Q1. During other times both chips are powered off to reduce total power dissipation.

The 2K PROM (U9) dissipates only standby-mode power until CE is active, so CE is made active only when A15 and A14 are both high. This is a compromise which requires no additional decoding.

The AND gates on the A8, A9, and A10 address lines of the 2K PROM force page 0 of the PROM to be accessed when the Cn00 page is referenced, regardless of what slot number is being used.

The flipflop U18-5 (the half of U18 whose output is on pin 5) is normally set, and is reset by I/O SELECT when the NIC is the selected card. It is clocked by the system clock once each cycle, and retains its current state until the decode PROM presents the DISABLE signal, which causes the flipflop to set.

The Apple][bus signals are relatively noisy, and consideration is taken in the design to minimize the effects of glitches. In particular, (1) the clock is received with a Schmitt-trigger input gate (74LS14 at U17-1), and (2) the deselect flipflop U18-5 is clocked with an ungated system clock so as to avoid sensitivity to address line glitches. (Note that this latter design represents a change in the -F version of the NIC compared to earlier revisions.)

The NIC Data Bus

The NIC has an internal data bus (ID0-ID7) to which the RAM, ROM, and PIAs are attached. The internal bus is gated onto the Apple peripheral bus by the bidirectional buffer U16, which is enabled for all I/O SELECT (256-byte block) references, and for I/O STROBE (2K-byte block) references when the NIC is the selected card (ENB is true). The buffer direction is controlled by the system R/W line.

The PIAs

Two 6520 (or 6521) Peripheral Interface Adapters are used to read and write the network data and control signals as well as various internal signals on the NIC. These are complex LSI devices, and the reader is referred to the PIA data sheet for details not given here. For reference purposes, figure 4 contains some summary information about the PIA taken from the data sheet.

Each PIA provides two 8-bit data ports and four control/status lines, most of which can be used for either input or output. Either rising or falling transitions of any of the status lines can be programmed to cause interrupts to the 6502 processor.

The PIAs are connected to the NIC internal data bus and appear to the processor as 4 memory locations which can be read and

written. For the NIC, those locations are as follows:

PIA #1	PIA #2	Function
CFF0	CFF4	Port A data/direction register
CFF1	CFF5	Port A control and status register
CFF2	CFF6	Port B data/direction register
CFF3	CFF7	Port B control and status register

The data/direction registers are either in data mode, allowing port data to be accessed, or in direction mode, allowing the port to be configured for input or output. The mode is controlled by bit 2 of the control and status register.

Each of the four PIA ports has two control lines associated with it. The "1" control lines (CA1 and CB1 for each PIA) can be used as input only. The "2" control lines (CA2 and CB2) can be programmed either as inputs or outputs, and are used as output on PIA #1 and as inputs on PIA #2.

Figure 3 shows the complete port and control/status bit assignments as used in the NIC.

Packet transmission by the PIAs

A summary of packet transmission over the network is useful in understanding the function of the PIAs. For more information, see the Electronics article referenced on the front page.

The network bus consists of 12 signals: 8 bidirectional data signals (BD0-BD7) and 4 handshake signals (HS1-HS4). The handshake lines are active low, and are used as timing and synchronization signals.

Whenever a station wishes to transmit, it makes sure the network is idle by checking that all data lines and HS3 are inactive (high). If so, it puts the address of the station it wishes to transmit to on the data bus, and makes HS3 active. It is the active edge of HS3 which triggers the address recognition logic on the NIC of the other station, and the software can detect that it is being addressed by looking at the latched result of a comparison between its own address and the data on the bus at the time HS3 became active.

HS3 remains active all during the time that the stations communicate with each other. They exchange data in "packets" of up to 256 data bytes plus length, type, and checksum information. During that transmission, handshake lines HS1 and HS2 are used to synchronize byte transfers; HS1 is used by the sender to indicate that the data is present, and HS2 is used by the receiver to indicate that the data has been accepted. (To save time, HS2 is not used in the inner loops for packet data.)

PIA #1

PIA #1 is primarily used to access network data. Port A is used to read network data through the transceivers U2/U5, and Port B is used to write the data. Note that by using separate ports it is possible to read the data while it is being written, which is necessary so that network collisions can be detected.

The network data from the transceivers are also presented to the 8-bit parallel comparator U1. The other comparator inputs are the address switches or shunts which have been set to the station address for the NIC. The result of the comparison is latched into flipflop U18-8 whenever the HS3 handshake line on the network is made active.

The four control lines of PIA #1 are used to read and write the HS1 and HS2 handshake lines. It is those lines which are used to synchronize individual byte transmission over the network.

PIA #2

PIA #2 is used primarily for reading and writing miscellaneous NIC control and status signals.

The handshake signals (HS1, HS2, HS3, HS4) and the address-match signal (MYADR) are inputs to port A of PIA #2 even though they are also connected to control inputs, because only transitions of control inputs can be detected. By reading port A the current status of those signals can be determined as well.

The network byte handshake signals (HS1 and HS2) are driven by the control lines of PIA #1, but the address/carrier signal (HS3)

and the currently unused signal (HS4) are driven by output bits on port B of PIA #2. Similarly the ROM/RAM mode is set by a port B output. The network transceivers (U2, U4, and U5) are also enabled by a port B output.

The address switches or shunts, in addition to being used by the comparator, must also be read by the processor. The address is read a bit at a time into bit 7 of port A of PIA #2. Multiplexor U3 is directed by three output bits of port B to choose one of the 8 address bits to be read. Thus all 8 bits can be read, one at a time, using only three output bits and one input bit.

The network transciever drivers are enabled or disabled by the HSENB output from port B of PIA #1. This line has a pullup so that the drivers are not enabled after the NIC has been reset but before it has been initialized by the software. (This represents a change on the -F revision.)

All control lines of PIA #2 (CA1, CB1, CA2, CB2) are programmed as inputs, and are used for the HS3 and HS4 handshake signals, the address-match signal, and the timer.

Figure 3 - PIA port and bit assignments

			Address
PIA #1	Port A:	Network data input	CFF0
	Port B:	Network data output	CFF2
	CA1:	HS1 transition input	CFF1
	CB1:	HS2 transition input	CFF3
	CA2:	HS2 output	CFF1
	CB2:	HS1 output	CFF3
PIA #2	Port A:	Miscellaneous inputs	CFF4

Bit 01: HS1 level input
Bit 02: HS2 level input
Bit 04: HS3 level input
Bit 08: HS4 level input
Bit 10: (unused)
Bit 20: Address comparator match
Bit 40: (unused)
Bit 80: Address switch data in

Port B: Miscellaneous outputs CFF6

Bits 07: Address switch bit select
Bit 08: ROM mode
Bit 10: not HS ENABLE
Bit 20: HS3 output
Bit 40: HS4 output
Bit 80: (unused)

CA1:	HS3 transition input	CFF5
CB1:	HS4 transition input	CFF7
CA2:	Clock transition input	CFF5
CB2:	Address comparator transition input	CFF7

Figure 4 (part 1) - PIA Reference Information

INTERNAL CONTROLS

There are six locations within the PIA accessible to the MPU data bus: two Peripheral Registers, two Data Direction Registers, and two Control Registers. Selection of these locations is controlled by the RS0 and RS1 inputs together with bit 2 in the Control Register, as shown in Table 1.

TABLE 1 – INTERNAL ADDRESSING

RS1	RS0	Control Register Bit CRA 2	Control Register Bit CRB 2	Location Selected
0	0	1	X	Peripheral Register A
0	0	0	X	Data Direction Register A
0	1	X	X	Control Register A
1	0	X	1	Peripheral Register B
1	0	X	0	Data Direction Register B
1	1	X	X	Control Register B

X Don't Care

INITIALIZATION

A low reset line has the effect of zeroing all PIA registers. This will set PA0-PA7, PB0-PB7, CA2 and CB2 as inputs, and all interrupts disabled. The PIA must be configured during the restart program which follows the reset.

Details of possible configurations of the Data Direction and Control Register are as follows.

DATA DIRECTION REGISTERS (DDRA and DDRB)

The two Data Direction Registers allow the MPU to control the direction of data through each corresponding peripheral data line. A Data Direction Register bit set at "0" configures the corresponding peripheral data line as an input; a "1" results in an output.

CONTROL REGISTERS (CRA and CRB)

The two Control Registers (CRA and CRB) allow the MPU to control the operation of the four peripheral control lines CA1, CA2, CB1 and CB2. In addition they allow the MPU to enable the interrupt lines and monitor the status of the interrupt flags. Bits 0 through 5 of the two registers may be written or read by the MPU when the proper chip select and register select signals are applied. Bits 6 and 7 of the two registers are read only and are modified by external interrupts occurring on control lines CA1, CA2, CB1 or CB2. The format of the control words is shown in Table 2.

TABLE 2 – CONTROL WORD FORMAT

	7	6	5	4	3	2	1	0
CRA	IRQA1	IRQA2	CA2 Control			DDRA Access	CA1 Control	
CRB	IRQB1	IRQB2	CB2 Control			DDRB Access	CB1 Control	

Data Direction Access Control Bit (CRA-2 and CRB-2) – Bit 2 in each Control register (CRA and CRB) allows selection of either a Peripheral Interface Register or the Data Direction Register when the proper register select signals are applied to RS0 and RS1.

Interrupt Flags (CRA-6, CRA-7, CRB-6, and CRB-7) - The four interrupt flag bits are set by active transitions of signals on the four Interrupt and Peripheral Control lines when those lines are programmed to be inputs. These bits cannot be set directly from the MPU Data Bus and are reset indirectly by a Read Peripheral Data Operation on the appropriate section.

TABLE 3 – CONTROL OF INTERRUPT INPUTS CA1 AND CB1

CRA-1 (CRB-1)	CRA-0 (CRB-0)	Interrupt Input CA1 (CB1)	Interrupt Flag CRA-7 (CRB-7)	MPU Interrupt Request $\overline{IRQA}$ ($\overline{IRQB}$)
0	0	↓ Active	Set high on ↓ of CA1 (CB1)	Disabled — $\overline{IRQ}$ remains high
0	1	↓ Active	Set high on ↓ of CA1 (CB1)	Goes low when the interrupt flag bit CRA-7 (CRB-7) goes high
1	0	↑ Active	Set high on ↑ of CA1 (CB1)	Disabled — $\overline{IRQ}$ remains high
1	1	↑ Active	Set high on ↑ of CA1 (CB1)	Goes low when the interrupt flag bit CRA-7 (CRB-7) goes high

Notes
1. ↑ indicates positive transition (low to high)
2. ↓ indicates negative transition (high to low)
3. The Interrupt flag bit CRA-7 is cleared by an MPU Read of the A Data Register, and CRB-7 is cleared by an MPU Read of the B Data Register
4. If CRA-0 (CRB-0) is low when an interrupt occurs (interrupt disabled) and is later brought high, $\overline{IRQA}$ ($\overline{IRQB}$) occurs after CRA-0 (CRB-0) is written to a "one".

Figure 4 (part 2) - PIA Reference Information

Control of CA1 and CB1 Interrupt Input Lines (CRA-0, CRB-0, CRA-1, and CRB-1) – The two lowest order bits of the control registers are used to control the interrupt input lines CA1 and CB1. Bits CRA-0 and CRB-0 are used to enable the MPU interrupt signals $\overline{IRQA}$ and $\overline{IRQB}$, respectively. Bits CRA-1 and CRB-1 determine the active transition of the interrupt input signals CA1 and CB1 (Table 3).

TABLE 4 – CONTROL OF CA2 AND CB2 AS INTERRUPT INPUTS
CRA5 (CRB5) is low

CRA-5 (CRB-5)	CRA-4 (CRB-4)	CRA-3 (CRB-3)	Interrupt Input CA2 (CB2)	Interrupt Flag CRA-6 (CRB-6)	MPU Interrupt Request $\overline{IRQA}$ ($\overline{IRQB}$)
0	0	0	↓ Active	Set high on ↓ of CA2 (CB2)	Disabled — $\overline{IRQ}$ remains high
0	0	1	↓ Active	Set high on ↓ of CA2 (CB2)	Goes low when the interrupt flag bit CRA-6 (CRB-6) goes high
0	1	0	↑ Active	Set high on ↑ of CA2 (CB2)	Disabled — $\overline{IRQ}$ remains high
0	1	1	↑ Active	Set high on ↑ of CA2 (CB2)	Goes low when the interrupt flag bit CRA-6 (CRB-6) goes high

Notes 1 ↑ indicates positive transition (low to high)

2 ↓ indicates negative transition (high to low)

3 The interrupt flag bit CRA-6 is cleared by an MPU Read of the A Data Register and CRB-6 is cleared by an MPU Read of the B Data Register

4 If CRA-3 (CRB-3) is low when an interrupt occurs (interrupt disabled) and is later brought high, $\overline{IRQA}$ ($\overline{IRQB}$) occurs after CRA-3 (CRB-3) is written to a "one"

TABLE 5 – CONTROL OF CB2 AS AN OUTPUT
CRB-5 is high

CRB-5	CRB-4	CRB-3	CB2	
			Cleared	Set
1	0	0	Low on the positive transition of the first E pulse following an MPU Write B Data Register operation	High when the interrupt flag bit CRB-7 is set by an active transition of the CB1 signal
1	0	1	Low on the positive transition of the first E pulse after an MPU Write "B" Data Register operation.	High on the positive edge of the first "E" pulse following an "E" pulse which occurred while the part was deselected.
1	1	0	Low when CRB-3 goes low as a result of an MPU Write in Control Register B	Always low as long as CRB-3 is low. Will go high on an MPU Write in Control Register B that changes CRB-3 to one
1	1	1	Always high as long as CRB-3 is high. Will be cleared when an MPU Write Control Register "B" results in clearing CRB-3 to zero	High when CRB-3 goes high as a result of an MPU Write into Control Register "B".

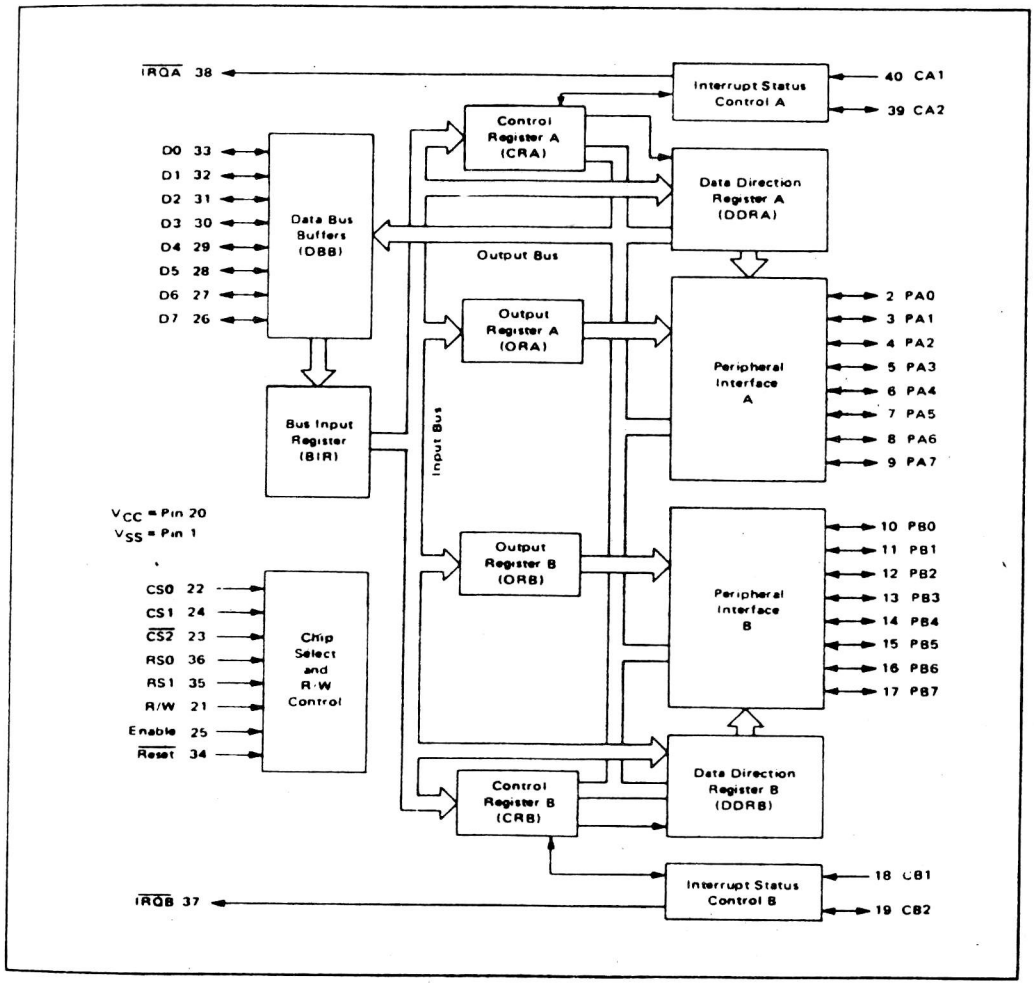
Figure 4 (part 3) - PIA Reference Information

Control of CA2 and CB2 Peripheral Control Lines (CRA-3, CRA-4, CRA-5, CRB-3, CRB-4, and CRB-5) – Bits 3, 4, and 5 of the two control registers are used to control the CA2 and CB2 Peripheral Control lines. These bits determine if the control lines will be an interrupt input or an output control signal. If bit CRA-5 (CRB-5) is low, CA2 (CB2) is an interrupt input line similar to CA1 (CB1) (Table 4). When CRA-5 (CRB-5) is high, CA2 (CB2) becomes an output signal that may be used to control peripheral data transfers. When in the output mode, CA2 and CB2 have slightly different characteristics (Tables 5 and 6).

TABLE 6 – CONTROL OF CA-2 AS AN OUTPUT
CRA-5 is high

CRA-5	CRA-4	CRA-3	CA2 Cleared	CA2 Set
1	0	0	Low on negative transition of E after an MPU Read "A" Data operation.	High when the interrupt flag bit CRA-7 is set by an active transition of the CA1 signal.
1	0	1	Low on negative transition of E after an MPU Read "A" Data operation.	High on the negative edge of the first "E" pulse which occurs during a deselect.
1	1	0	Low when CRA-3 goes low as a result of an MPU Write to Control Register "A".	Always low as long as CRA-3 is low. Will go high on an MPU Write to Control Register "A" that changes CRA-3 to "one".
1	1	1	Always high as long as CRA-3 is high. Will be cleared on an MPU Write to Control Register "A" that clears CRA-3 to a "zero".	High when CRA-3 goes high as a result of an MPU Write to Control Register "A".

EXPANDED BLOCK DIAGRAM



The Timer

The timer is a free-running 555 (U8) oscillating at about 10 hz. The output is connected to PIA #2 control input CA2, and can be polled by the processor or programmed to cause an interrupt.

Initial conditions

When the Apple has just been powered up, or after the RESET signal occurs, the NIC is in the following state:

-- The card is deselected (ENB is not asserted because U18-5 is preset)

-- The address match flipflop (U18-8) indicates no address match

-- All four PIA ports are configured as inputs

-- The network transceivers are disabled by pullup R5

-- ROM mode is asserted by pullup RN3 at U14-1

The Network Transceivers

The data lines of the network bus are driven and read by 26S11 bus transceivers U2 and U5. The drivers are non-inverting and the receivers are inverting, so the software must invert all input data. The receivers have a high switching threshold but no hysteresis. Hysteresis on the data lines is unnecessary because of the software strategy which guarantees sufficient settling time before the data is read.

The four handshake lines are driven and read by the MC3443 transceiver at U4. This transceiver inverts in both directions, and has 580 mv of input hysteresis.

All 12 network signal lines are driven by open collector high-current transistors, and there must be a resistive pullup somewhere on the network. Location RN2 on the NIC is provided

for the pullup, which is typically installed ONLY on the primary file server (station $FE) of each network.

The resistor pack is a 14-pin DIP with 13 equal value resistors connected to common pin 14. Pin 7 of RN2 is grounded so that a voltage divider resistor pack network can be used for special circumstances. In the normal case only pullups are used, and pin 7 of the resistor pack is clipped off to avoid unnecessary power dissipation.

The standard value for the network pullup is 680 ohms, and is sufficient for all network topologies with 1000′ of standard cable or less. Under special conditions with networks over 1000′, it may be acceptable to lower the pullup value, or to put more than one pullup on the system. Lower pullup values are especially useful in extending the network distance considerably beyond the 1000′ specification when low-resistance round cable is being used.

Note that if there is more than one pullup, ALL stations that have pullups MUST be powered on at ALL times for the network to operate. The total pullup value should not be less than 220 ohms or greater than 1K ohms.

The two standard network cables are: flat cables using IDC (insulation displacement) connectors, and round cables using DA-15 subminiature connectors. Figure 5 shows the signal names and pin numbering for both connector types. For more information about cabling, see Technical Note 6, "ClusterBus Cabling Information" (Rev 1, 9/4/81).

Figure 5 - Network signal names and connector numbering
sp 3

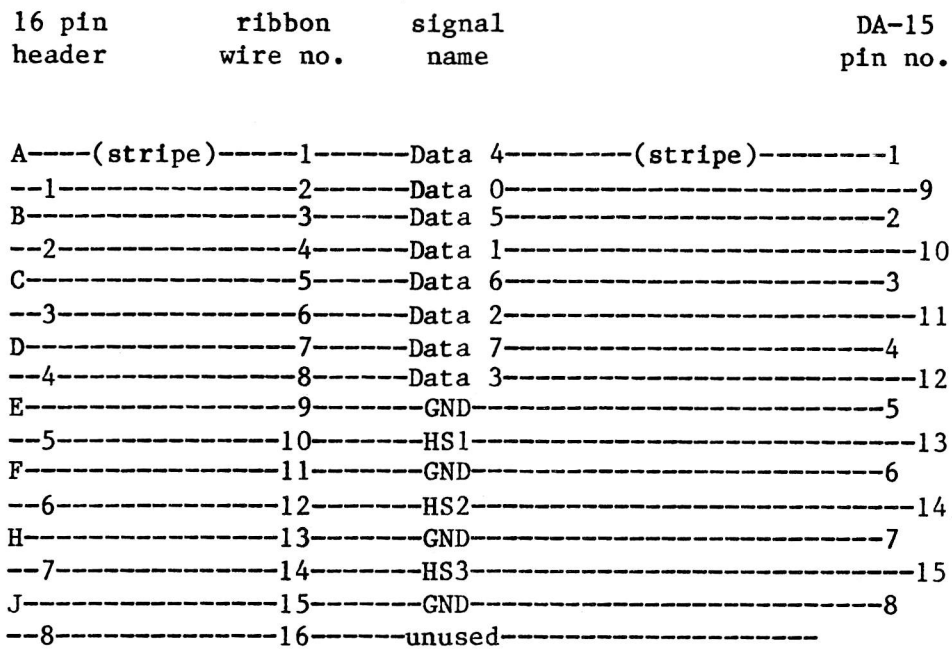


Figure 6 — NIC Schematic diagram